Felipe Gabriel de Souza Sphair

João Vitor Cidral Varpechoski

João Vitor Dadas de Oliveira

José Alcides Aguayo Mussy

Rafael Visnieski de Oliveira

Vitor Sério Bertoldi

RELAÇÃO DE ARTEFATOS

ESPECIFICAÇÃO DO PROJETO

-GrowHealthy-

Trabalho apresentado como requisito parcial para a disciplina de **Experiência Criativa – Projetando Soluções Computacionais**, do curso de Bacharelado em Engenharia de Software, da PUCPR.

Orientadoras:

Profa. Cristina Verçosa P. B. de Souza

Profa. Rosilene Fernandes

Curitiba

2023

SUMÁRIO

[ARTEFATO 1: Quadro “3 Objetivos” 3](#_Toc136796475)

[ARTEFATO 2: Quadro “é – não é – faz – não faz” 4](#_Toc136796476)

[ARTEFATO 3: Quadro “Visão de Produto”. 5](#_Toc136796477)

[ARTEFATO 4: Canvas PBB 6](#_Toc136796478)

[ARTEFATO 5: Relação de User Stories 12](#_Toc136796479)

[ARTEFATO 6: Modelo Relacional 16](#_Toc136796480)

[ARTEFATO 7: Demais Diagramas 17](#_Toc136796481)

[REFERÊNCIA BIBLIOGRÁFICAS 26](#_Toc136796482)

ÍNDICE DE ILUSTRAÇÕES

[Figura 1 – Quadro “3 Objetivos”. 3](#_Toc136796483)

[Figura 2 – Quadro “é – não é – faz – não faz”. 4](#_Toc136796484)

[Figura 3 – Quadro "Visão de Produto". Fonte: AGUIAR, F. 2018. 5](#_Toc136796485)

[Figura 4 – Canvas PBB: "Product Backlog Building". Fonte: AGUIAR, F. 2018. 11](#_Toc136796486)

[Figura 5 - User Stories e Critérios de Aceite do PBI Realizar cadastro aluno. Fonte: AGUIAR, F. 2018. 12](#_Toc136796487)

[Figura 8 – Diagramas de Atividades. 17](#_Toc136796488)

[Figura 9 – Diagramas de Máquina de Estados. 18](#_Toc136796489)

# ARTEFATO 1: Quadro “3 Objetivos”

|  |  |
| --- | --- |
| **ARTEFATO 1**: Quadro “3 Objetivos” | |
| **NOME DO PRODUTO**: | |
| **OBJETIVOS** | **DESCRIÇÃO** |
| 1 | Auxiliar profissionais de uma academia (Personal e nutricionista) a criar treinos e dietas de uma forma mais ágil e personalizada para cada aluno. |
| 2 | Manter o progresso dos alunos de uma academia mais organizado. |
| 3 | Entregar um software de qualidade para academias de pequeno porte a um preço acessível. |

Figura – Quadro “3 Objetivos”.

# ARTEFATO 2: Quadro “é – não é – faz – não faz”

|  |  |
| --- | --- |
| **ARTEFATO 2**: Quadro “é – não é – faz – não faz” | |
| **NOME DO PRODUTO**: | |
| **É**  - Alternativa mais acessível para academias procurando entregar uma plataforma tecnológica para seus alunos. | **Não é**  Uma aplicação de treinos por vídeo ou imagem |
| **Faz**  Ajuda os profissionais de uma academia a montar treinos com o uso de IA. | **Não faz**  Atualizações automáticas do progresso. |

Figura – Quadro “é – não é – faz – não faz”.

# ARTEFATO 3: Quadro “Visão de Produto”.

Para ... 🡪

É um ... 🡪

Que ... 🡪

Ao contrário de ... 🡪

O nosso produto ... 🡪

|  |  |
| --- | --- |
| **ARTEFATO 3**: Quadro “Visão de Produto” | |
| **NOME DO PRODUTO**: | |
| **CLIENTE-ALVO** | Academias de pequeno e médio porte. |
| **CATEGORIA-SEGMENTO** | Saúde e Fitness |
| **BENEFÍCIO-CHAVE** | Alunos de uma academia conseguem ter uma experiencia mais personalizada. |
| **DIFERENCIADO-CHAVE** | Uso de IA para auxiliar os profissionais que trabalham em uma academia |
| **META-VALOR.** | Facilitar a vida dos profissionais que trabalham em uma academia e personalizar a experiencia do aluno. |

Figura – Quadro "Visão de Produto". Fonte: AGUIAR, F. 2018.

# ARTEFATO 4: Canvas PBB

Linha do tempo

Descrição gerada automaticamenteDiagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente

Diagrama

Descrição gerada automaticamente

Figura 4 – Canvas PBB: "Product Backlog Building". Fonte: AGUIAR, F. 2018.

# ARTEFATO 5: Relação de User Stories

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 1 - PBI**: Realizar Cadastro Aluno (SPRINT 1) | |
| **COMO**: aluno  **POSSO**: Realizar meu cadastro de Aluno  **PARA**: preencher os dados do meu perfil | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: não possuo cadastro  **QUANDO**: preencho o formulário de cadastro com todas as informações.  **ENTÃO**: posso receber instruções personalizadas. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: já tenho cadastro.  **QUANDO**: preencho o formulário de cadastro com todas as informações.  **ENTÃO**: o sistema notifica que já existe uma conta cadastrada com meu cpf. |

Figura - User Stories e Critérios de Aceite do PBI Realizar cadastro aluno. Fonte: AGUIAR, F. 2018.

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 2 - PBI**: Postar treino personalizado do Aluno (SPRINT 1) | |
| **COMO**: Personal  **POSSO**: Realizar acesso ao meu perfil  **PARA**: Postar treino personalizado para o meu Aluno. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: exista ao menos um aluno associado ao meu perfil  **QUANDO**: acesso minha página de pedidos.  **ENTÃO**: postar treino personalizado para o aluno. |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: não exista alunos associados ao meu perfil.  **QUANDO**: acesso minha página de pedidos.  **ENTÃO**: o sistema mostra a página com notificação de que não há alunos associados a mim. |

Figura 6 - User Stories e Critérios de Aceite do PBI Postar treino personalizado do Aluno. Fonte: AGUIAR, F. 2018.

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 3 - PBI**: Realizar cadastro Personal (SPRINT 2) | |
| **COMO**: Personal  **POSSO**: Realizar meu cadastro como Personal  **PARA**: Postar treino personalizado para o meu Aluno. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: não tenho cadastro  **QUANDO**: preencho o formulário de cadastro como Personal  **ENTÃO**: o sistema realiza meu cadastro na base de dados |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: já tenho cadastro  **QUANDO**: preencho formulário de cadastro como Personal  **ENTÃO**: o sistema notifica que já existe um cadastro realizado no cpf e/ou cref informado |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 4 - PBI**: Realizar cadastro Nutricionista (SPRINT 2) | |
| **COMO**: Nutricionista  **POSSO**: Realizar meu cadastro como Nutricionista  **PARA**: Postar dieta personalizada para o meu Aluno. | |
| **Critério de**  **Aceite 1** | **DADO** **QUE**: não tenho cadastro  **QUANDO**: preencho o formulário de cadastro como Nutricionista  **ENTÃO**: o sistema realiza meu cadastro na base de dados |
| **Critério de**  **Aceite 2** | **DADO** **QUE**: já tenho cadastro  **QUANDO**: preencho formulário de cadastro como Nutricionista  **ENTÃO**: o sistema notifica que já existe um cadastro realizado no cpf e/ou crn informado |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 5 - PBI**: Consultar orientação de treino personalizado do aluno via IA (SPRINT 2) | |
| **COMO**: Personal  **POSSO**: Consultar orientação de treino personalizado do aluno via IA como Personal  **PARA**: IA para auxiliar a montar treinos | |
| **Critério de**  **Aceite 1** | **DADO QUE:** acesso minha página de pedidos  **QUANDO:** abro a funcionalidade de montar treinos e a comunicação com a IA está ativa  **ENTÃO:** o sistema abre a página de postar treino com o campo de texto do treino preenchido com o treino gerado pela IA |
| **Critério de**  **Aceite 2** | **DADO QUE:** acesso minha página de pedidos  **QUANDO:** abro a funcionalidade de montar treinos e a comunicação com a IA está inativa  **ENTÃO:** o sistema notifica que a comunicação com a IA está inativa e permite que eu escreva o treino manualmente |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 6 - PBI**: Montar treino personalizado do aluno (SPRINT 2) | |
| **COMO**: Personal  **POSSO**: Montar treino personalizado do aluno  **PARA**: Postar treino personalizado do aluno | |
| **Critério de**  **Aceite 1** | **DADO QUE:** acesso a funcionalidade de montar treinos  **QUANDO:** abro a funcionalidade de montar treinos e a comunicação com a IA está ativa  **ENTÃO:** reviso o treino gerado apresentado pela IA no campo de texto e o posto para o aluno. |
| **Critério de**  **Aceite 2** | **DADO QUE:** acesso a funcionalidade de montar treinos  **QUANDO:** abro a funcionalidade de montar treinos e a comunicação com a IA está inativa  **ENTÃO:** o sistema notifica que a comunicação com a IA está inativa e permite que eu escreva o treino manualmente |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 7 - PBI**: Consultar orientação de dieta personalizada do aluno via IA (SPRINT 2) | |
| **COMO**: Nutricionista  **POSSO**: Consultar orientação de dieta personalizada do aluno via IA como Personal  **PARA**: IA para auxiliar a montar dieta | |
| **Critério de**  **Aceite 1** | **DADO QUE:** acesso minha página de pedidos  **QUANDO:** abro a funcionalidade de montar dietas e a comunicação com a IA está ativa  **ENTÃO:** o sistema abre a página de montar dieta com o campo de texto da dieta preenchido com a dieta gerada pela IA |
| **Critério de**  **Aceite 2** | **DADO QUE:** acesso minha página de pedidos  **QUANDO:** abro a funcionalidade de montar dietas e a comunicação com a IA está inativa  **ENTÃO:** o sistema notifica que a comunicação com a IA está inativa e permite que eu escreva a dieta manualmente |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 8 - PBI**: Montar dieta personalizada do Aluno (SPRINT 2) | |
| **COMO**: Nutricionista  **POSSO**: Montar dieta personalizada do aluno  **PARA**: Postar dieta personalizada do aluno | |
| **Critério de**  **Aceite 1** | **DADO QUE:** acesso a funcionalidade de montar dietas  **QUANDO:** abro a funcionalidade de montar dietas e a comunicação com a IA está ativa  **ENTÃO:** reviso a dieta gerada apresentada pela IA no campo de texto e a posto para o aluno. |
| **Critério de**  **Aceite 2** | **DADO QUE:** acesso a funcionalidade de montar dietas  **QUANDO:** abro a funcionalidade de montar dietas e a comunicação com a IA está inativa  **ENTÃO:** o sistema notifica que a comunicação com a IA está inativa e permite que eu escreva a dieta manualmente. |

|  |  |
| --- | --- |
| **HISTÓRIA DO USUÁRIO 9 - PBI**: Postar dieta personalizada do Aluno (SPRINT 2) | |
| **COMO**: Nutricionista  **POSSO**: Realizar postagem de dieta do aluno  **PARA**: Postar dieta personalizada do aluno | |
| **Critério de**  **Aceite 1** | **DADO QUE:** exista ao menos um aluno associado ao meu perfil  **QUANDO:** acesso minha página de pedidos  **ENTÃO:** posso postar dieta personalizado para o aluno |
| **Critério de**  **Aceite 2** | **DADO QUE:** não exista alunos associados ao meu perfil  **QUANDO:** acesso minha página de pedidos  **ENTÃO:** o sistema mostra página com notificação de que não há alunos associados a mim |

# ARTEFATO 6: Modelo Relacional

|  |
| --- |
| **ARTEFATO 6:** Modelo Relacional |
|  |

Figura 7 –Modelo Relacional gerado por engenharia reversa (MySQL Workbench).

# ARTEFATO 7: Demais Diagramas

1. Diagrama de Atividades (da disciplina de Criação de Modelos Computacionais)

|  |
| --- |
| **ARTEFATO 8.1:** Diagrama de Atividades |
|  |

Figura – Diagramas de Atividades.

1. Diagrama de Máquina de Estado (da disciplina de Criação de Modelos Computacionais)

|  |
| --- |
| **ARTEFATO 8.2:** Diagrama de Máquinas de Estados |
|  |

Figura – Diagramas de Máquina de Estados.

**CÓDIGO SQL**

-- MySQL Workbench Forward Engineering

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';

-- -----------------------------------------------------

-- Schema mydb

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema growhealthy

-- -----------------------------------------------------

-- -----------------------------------------------------

-- Schema growhealthy

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `growhealthy` DEFAULT CHARACTER SET utf8mb4 COLLATE utf8mb4\_0900\_ai\_ci ;

USE `growhealthy` ;

-- -----------------------------------------------------

-- Table `growhealthy`.`academia`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`academia` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nomeFantasia` VARCHAR(100) NULL DEFAULT NULL,

`cnpj` VARCHAR(14) NULL DEFAULT NULL,

`login` VARCHAR(50) NULL DEFAULT NULL,

`senha` VARCHAR(50) NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `cnpj\_UNIQUE` (`cnpj` ASC) VISIBLE)

ENGINE = InnoDB

AUTO\_INCREMENT = 4

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----------------------------------------------------

-- Table `growhealthy`.`personal`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`personal` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(100) NULL DEFAULT NULL,

`cref` VARCHAR(20) NULL DEFAULT NULL,

`login` VARCHAR(50) NULL DEFAULT NULL,

`senha` VARCHAR(50) NULL DEFAULT NULL,

`celular` VARCHAR(14) NULL DEFAULT NULL,

`email` VARCHAR(45) NULL DEFAULT NULL,

`dt\_nasc` VARCHAR(45) NULL DEFAULT NULL,

`genero` VARCHAR(45) NULL DEFAULT NULL,

`cpf` VARCHAR(11) NULL DEFAULT NULL,

`academia\_id` INT NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `cref` (`cref` ASC) VISIBLE,

INDEX `academia\_id` (`academia\_id` ASC) VISIBLE,

CONSTRAINT `personal\_ibfk\_1`

FOREIGN KEY (`academia\_id`)

REFERENCES `growhealthy`.`academia` (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 12

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----------------------------------------------------

-- Table `growhealthy`.`nutricionista`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`nutricionista` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(100) NULL DEFAULT NULL,

`crn` VARCHAR(20) NULL DEFAULT NULL,

`login` VARCHAR(50) NULL DEFAULT NULL,

`senha` VARCHAR(50) NULL DEFAULT NULL,

`celular` VARCHAR(14) NULL DEFAULT NULL,

`email` VARCHAR(45) NULL DEFAULT NULL,

`dt\_nasc` VARCHAR(45) NULL DEFAULT NULL,

`genero` VARCHAR(45) NULL DEFAULT NULL,

`cpf` VARCHAR(11) NULL DEFAULT NULL,

`academia\_id` INT NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `crn` (`crn` ASC) VISIBLE,

INDEX `academia\_id` (`academia\_id` ASC) VISIBLE,

CONSTRAINT `nutricionista\_ibfk\_1`

FOREIGN KEY (`academia\_id`)

REFERENCES `growhealthy`.`academia` (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 4

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----------------------------------------------------

-- Table `growhealthy`.`aluno`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`aluno` (

`id` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(100) NULL DEFAULT NULL,

`cpf` VARCHAR(11) NULL DEFAULT NULL,

`login` VARCHAR(50) NULL DEFAULT NULL,

`senha` VARCHAR(50) NULL DEFAULT NULL,

`celular` VARCHAR(14) NULL DEFAULT NULL,

`email` VARCHAR(45) NULL DEFAULT NULL,

`dt\_nasc` VARCHAR(45) NULL DEFAULT NULL,

`genero` VARCHAR(45) NULL DEFAULT NULL,

`peso` FLOAT NULL DEFAULT NULL,

`altura` FLOAT NULL DEFAULT NULL,

`restricoesFisicas` TEXT NULL DEFAULT NULL,

`restricoesAlimentares` TEXT NULL DEFAULT NULL,

`personal\_id` INT NULL DEFAULT NULL,

`nutricionista\_id` INT NULL DEFAULT NULL,

`academia\_id` INT NULL DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE INDEX `cpf` (`cpf` ASC) VISIBLE,

UNIQUE INDEX `email\_UNIQUE` (`email` ASC) VISIBLE,

INDEX `academia\_id` (`academia\_id` ASC) VISIBLE,

INDEX `personal\_id` (`personal\_id` ASC) VISIBLE,

INDEX `nutricionista\_id` (`nutricionista\_id` ASC) VISIBLE,

CONSTRAINT `aluno\_ibfk\_1`

FOREIGN KEY (`academia\_id`)

REFERENCES `growhealthy`.`academia` (`id`),

CONSTRAINT `aluno\_ibfk\_2`

FOREIGN KEY (`personal\_id`)

REFERENCES `growhealthy`.`personal` (`id`),

CONSTRAINT `aluno\_ibfk\_3`

FOREIGN KEY (`nutricionista\_id`)

REFERENCES `growhealthy`.`nutricionista` (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 7

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----------------------------------------------------

-- Table `growhealthy`.`dieta`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`dieta` (

`id` INT NOT NULL AUTO\_INCREMENT,

`descricao` TEXT NULL DEFAULT NULL,

`dataCriacao` DATE NULL DEFAULT NULL,

`ativo` TINYINT NULL DEFAULT '1',

`aluno\_id` INT NULL DEFAULT NULL,

`nutricionista\_id` INT NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `aluno\_id` (`aluno\_id` ASC) VISIBLE,

INDEX `nutricionista\_id` (`nutricionista\_id` ASC) VISIBLE,

CONSTRAINT `dieta\_ibfk\_1`

FOREIGN KEY (`aluno\_id`)

REFERENCES `growhealthy`.`aluno` (`id`),

CONSTRAINT `dieta\_ibfk\_2`

FOREIGN KEY (`nutricionista\_id`)

REFERENCES `growhealthy`.`nutricionista` (`id`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

-- -----------------------------------------------------

-- Table `growhealthy`.`treino`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `growhealthy`.`treino` (

`id` INT NOT NULL AUTO\_INCREMENT,

`descricao` TEXT NULL DEFAULT NULL,

`dataCriacao` DATE NULL DEFAULT NULL,

`ativo` TINYINT NULL DEFAULT '1',

`aluno\_id` INT NULL DEFAULT NULL,

`personal\_id` INT NULL DEFAULT NULL,

PRIMARY KEY (`id`),

INDEX `aluno\_id` (`aluno\_id` ASC) VISIBLE,

INDEX `personal\_id` (`personal\_id` ASC) VISIBLE,

CONSTRAINT `treino\_ibfk\_1`

FOREIGN KEY (`aluno\_id`)

REFERENCES `growhealthy`.`aluno` (`id`),

CONSTRAINT `treino\_ibfk\_2`

FOREIGN KEY (`personal\_id`)

REFERENCES `growhealthy`.`personal` (`id`))

ENGINE = InnoDB

AUTO\_INCREMENT = 3

DEFAULT CHARACTER SET = utf8mb4

COLLATE = utf8mb4\_0900\_ai\_ci;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

**LINKS:**

Gestão de projeto:

<https://trello.com/b/DVu6u1fX/untitled-board>

Controle de versionamento:

<https://github.com/Josemussy/GrowHealthy>

# REFERÊNCIA BIBLIOGRÁFICAS

AGUIAR, F. **Product backlog building: concepção de um product backlog efetivo**. 2018. Disponível em: <https://speakerdeck.com/fabiogr/product-backlog-building>. Acesso em: 10 fevereiro 2022.

AGUIAR, F. **PBB\_Canvas Template**. 2018. Disponível em: <http://www.productbacklogbuilding.com/canvas/PBB_Canvas.pdf>. Acesso em 10 de fevereiro de 2022.

SCHWABER, K.; SUTHERLAND, J. **Guia do SCRUM - o guia definitivo para o Scrum: as regras do jogo**. 2020. Disponível em: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Portuguese-European.pdf>. Acesso em: 10 fevereiro 2022.